WINDOWS

วิรัช วินิจวัฒนาวงษ์

การแชร์ข้อมูลระหว่างแอพพลิเคชัน : OLE

พบกับความสามารถของการตรึงข้อมูลในแอพพลิเคชันที่ต่าง ๆ กัน เพื่อให้เอกสารข้อมูลมีความหลากหลายมากขึ้น

Object Linking and Embedding (OLE) เป็นระบบการแชร์ข้อมูลแบบใหม่ของวินโดวส์ 3.1 ซึ่งจะอนุญาตให้เรารวมออปเจ็กต์ เช่น เท็กซ์หรือกราฟิกจากแอพพลิเคชันที่ต่างกันมาไว้ในเอกสารหนึ่ง ๆ ตัวอย่างเช่น เราสามารถสร้างเอกสารในเวิร์ดฟอร์วินโดวส์ที่มีชาร์ตของเอ็กเซล มีรูปภาพของเพนต์บรัชได้กี่ชาร์ตหรือกี่รูปก็ได้ นอกเหนือจากเท็กซ์ของเวิร์ดฟอร์วินโดวส์เองแล้ว ชาร์ตและรูปภาพเหล่านี้สามารถที่จะตรึง (Embedded) เอาไว้ตรงจุดบริเวณต่าง ๆ ที่เราเลือกในเอกสารของเวิร์ดฟอร์วินโดวส์

ความสามารถจริง ๆ ของ OLE นั้นจะอยู่ในตอนที่เราจำเป็นต้องแก้ไขสิ่งที่ตรึงเอาไว้ในเอกสาร แทนที่เราจะต้องออกจากเวิร์ดฟอร์วินโดวส์และไปเริ่มต้นแอพพลิเคชันที่สร้างสิ่งที่ตรึงนี้ แต่เราไม่ต้องออก เราสามารถดับเบิ้ลคลิ๊กบนออปเจ็กต์หรือสิ่งที่ตรึงไว้ในเวิร์ดฟอร์วินโดวส์ วินโดวส์จะเปิด

แอพพลิเคชันที่ใช้สร้างออปเจ็กต์ที่ถูกตรึงนั้น และแสดงออปเจ็กต์เพื่อให้เราแก้ไขได้ จากนั้นเราก็ทำการเปลี่ยนแปลงแก้ไขในสิ่งที่ต้องการเมื่อเรากลับไปยังเวิร์ดฟอร์วินโดวส์ การเปลี่ยนแปลงที่เกิดขึ้นกับออปเจ็กต์ที่แก้ไข จะเกิดขึ้นกับออปเจ็กต์ที่ตรึงไว้นี้ด้วย เวิร์ดฟอร์วินโดวส์จะปรับปรุงและแสดงลักษณะใหม่ที่เปลี่ยนแปลงให้เห็น

ประโยชน์อีกอย่างหนึ่งของ OLE คือ ความง่ายในการใช้ ซึ่งเราสามารถแทรกและแก้ไขออปเจ็กต์ที่นำมาตรึงไว้นี้ได้ง่าย ขั้นตอนทั้งหมดก็เพียงแต่ใช้เม้าส์เป็นตัวจัดการ อันที่จริงแล้วการตรึงออปเจ็กต์แบบ OLE ไม่ได้มีความซับซ้อนอะไรเลย ก็เหมือน ๆ กับที่เราใช้คำสั่ง Copy/Edit Paste กันอยู่บ่อย ๆ นั่นเอง

OLE อยู่บนระบบเมสเสจของ DDE ซึ่งจะถูกส่งผ่านข่าวสารข้อมูลเกี่ยวกับออปเจ็กต์ระหว่างแอพพลิเคชัน แอพพิลเคชันที่สามารถรับออปเจ็กต์จากแอพพลิเคชันอื่นจะเรียกว่าเป็นไคล์เอ็นต์ และแอพพลิเคชันที่เป็นผู้ให้ออปเจ็กต์จะเรียกเป็นเซอร์ฟเวอร์

อย่างไรก็ตาม ไม่ใช่ทุกแอพพลิเคชันที่จะมีบทบาททั้งสองอย่างคือไคล์เอ็นต์และเซอร์ฟเวอร์ ตัวอย่างเช่น แอพพลิเคชันของวินโดวส์ 3.1 ได้แก่ Write, Cardfile, Paintbrush และ Sound Recorder แอพพลิเคชัน Write และ Cardfile ใช้เป็นไคล์เอ็นต์ได้เท่านั้น (ขอออปเจ็กต์จากแอพพลิเคชันอื่น) ส่วน Paintbrush และ Sound Recorder ใช้เป็นเซอร์ฟเวอร์เท่านั้น (ให้ออปเจ็กต์กับคนอื่น)

ไม่ใช่ว่าแอพพลิเคชันจะสามารถใช้คุณสมบัติ OLE ได้โดยอัตโนมัติภายใต้วินโดวส์ 3.1 มันจะต้องถูกพัฒนาให้ใช้กับ OLE หรือพูดง่าย ๆ ว่าในตัวมันจะต้องมีคุณสมบัติของการใช้ OLE อยู่ด้วย ซึ่งมาระยะหลังผู้ผลิตซอฟต์แวร์จะใส่คุณสมบัตินี้ไว้กับแอพพลิเคชันของตัวเองส่วนใหญ่ แอพพลิเคชันดัง ๆ ที่ใช้คุณสมบัติ OLE ได้ก็อย่างเช่น เอ็กเซลและเวิร์ดฟอร์วินโดวส์ ทั้งคู่สามารถเป็นได้ทั้งไคล์เอ็นต์และเซอร์ฟเวอร์

ทำงานกันอย่างไร

การส่งผ่านออปเจ็กต์จากเซอร์ฟเวอร์ไปยังไคล์เอ็นต์และปรับปรุงออปเจ็กต์ที่มีการเปลี่ยนแปลงนั้น แอพพลิเคชันจะเกี่ยวข้องกับ DDLs (Dynamic Link Libraries) สองตัวคือ OLECLI.DLL (สำหรับไคล์เอ็นต์) และ OLESRV.DLL (สำหรับเซอร์ฟเวอร์) การใช้ฟังก์ชันใน DDLs ของ OLE ก็เพื่อการส่งเมสเสจไปมาระหว่างแอพพลิเคชันนั้นเอง

เมื่อเราร้องขอเกี่ยวกับการเปลี่ยนแปลงของออปเจ็กต์ที่ตรึงไว้ในแอพพลิเคชันที่เป็นไคล์เอ็นต์ แอพพลิเคชันไคล์เอ็นต์จะทำการเรียกไลบรารีไคล์เอ็นต์ OLECLI.DLL ไลบรารีนี้จะส่งเมสเสจที่เกี่ยวกับการร้องขอไปยังไลบรารีเซอร์ฟเวอร์ OLESRV.DLL จากนั้นไลบรารีเซอร์ฟเวอร์ก็จะตอบสนองโดยการเปิดแอพพลิเคชันที่เป็นเซอร์ฟเวอร์ และแสดงออปเจ็กต์เพื่อให้เราเปลี่ยนแปลงแก้ไข เมื่อเราแก้ไขออปเจ็กต์ ไลบรารีเซอร์ฟเวอร์จะส่งเมสเสจไปยังไลบรารีไคล์เอ็นต์เพื่อบอกให้ทราบว่ามีการเปลี่ยนแปลงเกิดขึ้น จากนั้นไลบรารีไคล์เอ็นต์จะปรับปรุงสิ่งที่เปลี่ยนแปลงในแอพพลิเคชันไคล์เอ็นต์

DDLs อีกตัวหนึ่งที่สำคัญคือ SHELL.DLL ไลบรารีนี้อนุญาตให้เราลงทะเบียนแอพพลิเคชันในทะเบียนดาต้าเบสของวินโดวส์ วินโดวส์ใช้ดาต้าเบสนี้พิจารณาว่าแอพพลิเคชันใดที่ใช้คุณสมบัติ OLE ได้และระดับที่สามารถใช้ได้ นอกจากนี้ยังใช้พิจารณาว่าแอพพลิเคชันใดสนับสนุนคุณสมบัติแดร็กแอนด์ดร็อบ และบันทึกไฟล์ที่เกี่ยวข้องที่เราสร้างขึ้นในไฟล์แมนเนเจอร์

การตรึงและการเชื่อมโยง

ออปเจ็กต์ที่ส่งผ่านกันมาในลักษณะของ OLE อาจจะเป็นเพียงเซลล์เดียวในสเปรดชีต หรือทั้งเท็กซ์ไฟล์ก็ได้ ออปเจ็กต์นี้สามารถนำมาตรึงหรือเชื่อมโยงในไฟล์ข้อมูลของไคล์เอ็นต์ ซึ่งการตรึงและการเชื่อมโยงจะให้ลักษณะที่แตกต่างกัน

การตรึงออปเจ็กต์ในไฟล์ข้อมูลไคล์เอ็นต์จะกลายเป็นส่วนหนึ่งของไฟล์ไปเลย ตัวอย่างเช่น ถ้าเราตรึงย่านเซลล์สเปรดชีตของเอ็กเซลในเอกสารของเวิร์ดฟอร์วินโดวส์ เมื่อออปเจ็กต์ถูกตรึงไว้ การอ้างอิงของเอ็กเซลซึ่งตอนนี้จะเป็นแอพพลิเคชันที่เป็นเซอร์ฟเวอร์จะตรึงด้วยกันกับข้อมูลของมันเอง ซึ่งรวมถึงสูตรที่ให้ข้อมูลด้วย ดังนั้นออปเจ็กต์ตัวมันเองจะถูกตรึงไว้ในไฟล์ข้อมูลของไคล์เอ็นต์ และจะถูกเซฟลงในดิสก์ด้วยกันกับไฟล์นี้

เมื่อเราดับเบิ้ลคลิ๊กบนออปเจ็กต์ที่ตรึงไว้นี้ในแอพพลิเคชันที่เป็นไคล์เอ็นต์ วินโดวส์จะเปิดแอพพลิเคชันที่เป็นเซอร์ฟเวอร์และแสดงออปเจ็กต์เพื่อให้เราแก้ไข ซึ่งก็คือการเปิดแอพพลิเคชันเซอร์ฟเวอร์และโหลดข้อมูลจากไฟล์ข้อมูลของไคล์เอ็นต์มาไว้ในแอพพลิเคชันของเซอร์ฟเวอร์นั่นเอง จากนี้เราก็

สามารถแก้ไขเปลี่ยนแปลงออปเจ็กต์และให้ไฟล์ข้อมูลของไคล์เอ็นต์ปรับปรุงเพื่อตอบสนองต่อการเปลี่ยนแปลงที่เกิดขึ้น

ในทางกลับกัน ลักษณะของการเชื่อมโยงออปเจ็กต์ ตัวออปเจ็กต์จะเป็นไฟล์ข้อมูลแยกต่างหาก ซึ่งอยู่ในความดูแลของแอพพลิเคชันเซอร์ฟเวอร์ การเชื่อมโยงกับออปเจ็กต์เท่านั้นที่จะถูกเซฟไว้ในไฟล์ข้อมูลของไคล์เอ็นต์ โดยจะไม่เซฟตัวข้อมูลออปเจ็กต์ลงในไฟล์นี้ การแสดงของออปเจ็กต์ที่เป็นการเชื่อมโยงในไคล์เอ็นต์จะเหมือนกันกับออปเจ็กต์ที่เป็นการตรึง เราสามารถดับเบิ้ลคลิ๊กบนออปเจ็กต์ที่เชื่อมโยงในแอพพลิเคชันไคล์เอ็นต์เพื่อให้วินโดวส์เปิดแอพพลิเคชันเซอร์ฟเวอร์และแสดงออปเจ็กต์เพื่อแก้ไขได้เช่นเดียวกันกับออปเจ็กต์ที่เป็นแบบการตรึง การเปลี่ยนแปลงที่เราทำจะเกิดขึ้นกับแอพพลิเคชันไคล์เอ็นต์โดยอัตโนมัติเช่นกัน

ถ้าต้องการให้การเปลี่ยนแปลงเป็นแบบถาวร เราต้องเซฟออปเจ็กต์ในดิสก์จากแอพพลิเคชันที่เป็นเซอร์ฟเวอร์ ถ้ามีเอกสารที่เราสร้างขึ้นด้วยแอพพลิเคชันไคล์เอ็นต์ที่มีคุณสมบัติ OLE เชื่อมโยงกับออปเจ็กต์นี้ ทุกเอกสารนั้น ๆ จะมีการปรับปรุงด้วยเมื่อมีการเปลี่ยนแปลงเกิดขึ้น ดังนั้นการเชื่อมโยงออปเจ็กต์แทนที่จะตรึงไว้ แต่จะเป็นประโยชน์เมื่อเราต้องการให้ออปเจ็กต์หนึ่งใช้กับหลาย ๆ เอกสารที่ต้องการออปเจ็กต์ที่เหมือนกันนี้

แสดงในลักษณะแพคเกจ

ออปเจ็กต์สามารถเชื่อมโยงหรือตรึงไว้ในเอกสารของแอพพลิเคชันที่เป็นไคล์เอ็นต์ได้สองแบบคือ แบบแรกเป็นการให้แสดงออปเจ็กต์ตัวมันเองเลย เช่น กราฟของเอ็กเซลหรือภาพบิตแม็พของเพนต์บรัชในเอกสารเวิร์ดฟอร์วินโดวส์- อีกแบบหนึ่งเป็นการแสดงในลักษณะไอคอน ซึ่งเป็นสัญลักษณ์ที่ใช้แทนออปเจ็กต์ ไอคอนที่ตรึงไว้นี้เรียกว่าแพคเกจ (package) เมื่อเราดับเบิ้ลคลิ๊กบนแพคเกจ วินโดวส์จะเปิดแอพพลิเคชันที่ใช้สร้างออปเจ็กต์และแสดงมัน ทูลส์ส่วนใหญ่สำหรับสร้างแพคเกจจะใช้ไลบรารี SHELL.DLL ไลบรารีนี้สนับสนุนการสร้างแพคเกจได้หลายวิธี วิธีหนึ่งก็โดยใช้แอพพลิเคชัน Object Packager ที่มากับวินโดวส์ 3.1

การตรึงออปเจ็กต์

การดำเนินการเกี่ยวกับ OLE ในแอพพลิเคชัน มีวิธีที่ง่าย ๆ โดยใช้คำสั่งที่เกี่ยวข้องกับ OLE ในแอพพลิเคชันที่เป็นไคล์เอ็นต์และเซอร์ฟเวอร์นั้น เราจะดูตัวอย่างของโปรแกรม Write กับ Paintbrush และแอพพลิเคชันเอ็กเซลกับเวิร์ดฟอร์วินโดวส์

Write และ Paintbrush

เราจะตรึงออปเจ็กต์ซึ่งเป็นภาพที่วาดในเพนต์บรัชลงในเอกสารของโปรแกรม Write เราสามารถเริ่มต้นจาก Write หรือ Paintbrush ก่อนก็ได้ เราจะมาเริ่มกันที่ Write ก่อน โดยจะแทรกออปเจ็กต์ที่จะตรึงไว้ เมื่อออปเจ็กต์ถูกตรึงแล้ว เราจะเรียก Paintbrush จากในโปรแกรม Write และทำการแก้ไขออปเจ็กต์

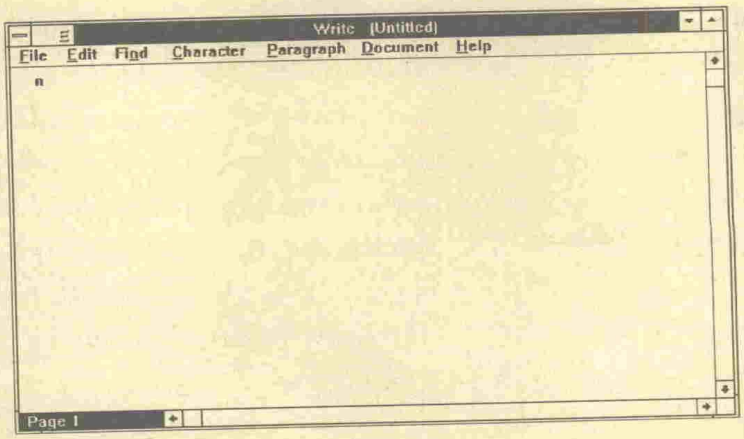
ขั้นตอนในการตรึงเมื่อเริ่มต้นจากโปรแกรม Write กระทำดังนี้

1. เริ่มต้น Write โดยดับเบิ้ลคลิ๊กที่ไอคอน Write ในกรุ๊ป Accessories จะปรากฏดังรูปที่ 1 จากนั้นให้วางเคอร์เซอร์ตรงตำแหน่งที่ต้องการให้ออปเจ็กต์ปรากฏ

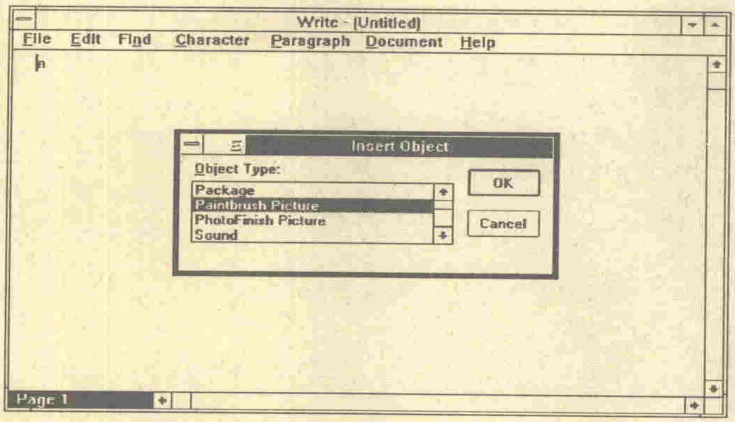
2. บนเมนู Edit เลือกคำสั่ง Insert Object จะปรากฏไดอะล็อกบ็อกซ์ดังรูปที่ 2 ซึ่งแสดงชื่อของแอพพลิเคชันเซอร์ฟเวอร์ที่ใช้คุณสมบัติ OLE ซึ่งลงทะเบียนไว้แล้วในระบบของเรา

3. เลือก Paintbrush Picture ในรายการ และเลือก OK

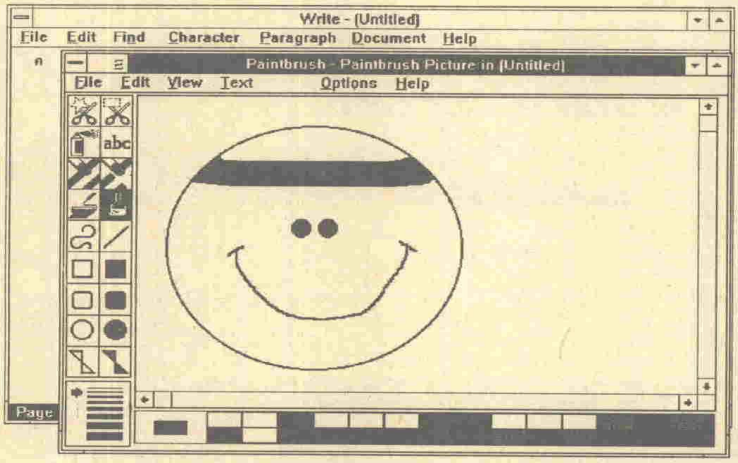
วินโดวส์จะเปิดโปรแกรม Paintbrush เมื่อเพนต์บรัชถูกเปิดขึ้นมาแล้ว ออปเจ็กต์ตอนนี้ก็คือภาพที่ว่าง ๆ อยู่ เรามีทางเลือกอยู่สองอย่างในการกำหนดออปเจ็กต์ที่จะตรึง อย่างแรกก็โดยการวาดรูปภาพใหม่ขึ้นเอง อย่างที่สองคือ



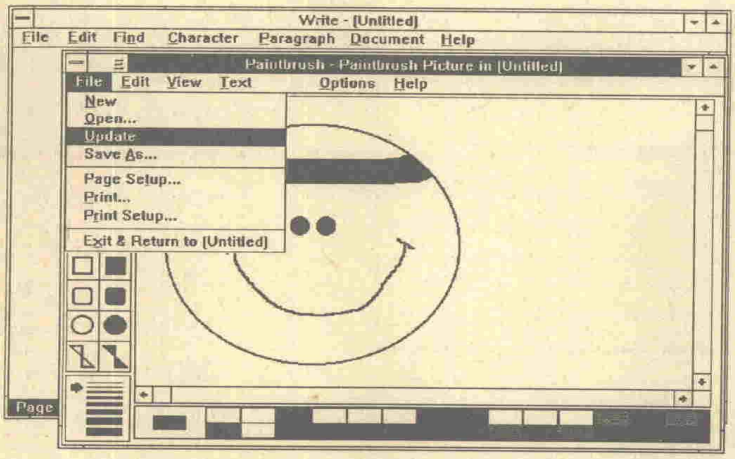
รูปที่ 1
เริ่มต้นโปรแกรม Write สำหรับออปเจ็กต์ที่จะมาตรึงไว้



รูปที่ 2
เลือกแอพพลิเคชันเซอร์ฟเวอร์ (Paintbrush) ที่ใช้สร้างออปเจ็กต์ที่จะนำมาตรึง



รูปที่ 3
เมื่อวินโดวส์เปิดเซอร์ฟเวอร์ (Paintbrush) ขึ้นมา ให้โหลดหรือวาดรูปออปเจ็กต์ที่ต้องการ



รูปที่ 4
ปรับปรุงออปเจ็กต์ที่มีการเปลี่ยนแปลง

โหลดรูปภาพที่มีอยู่แล้วโดยใช้คำสั่ง Paste From บนเมนู Edit ในตัวอย่างนี้เราจะวาดรูปขึ้นเองโดยกระทำดังนี้

1. วาดรูปอะไรก็ได้ลงในเพนต์บรัช ตัวอย่างแสดงดังในรูปที่ 3

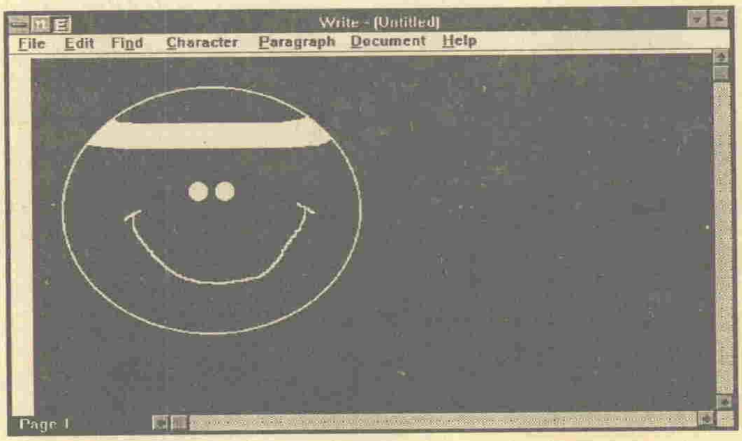
2. บนเมนู File เลือกคำสั่ง Update ดังรูปที่ 4 เพื่อปรับปรุงออปเจ็กต์ (รูปที่วาดขึ้นมาใหม่)

3. บนเมนู File เลือกคำสั่ง Exit & Return to (Untitled) เพื่อกลับคืนสู่ Write อีกครั้ง

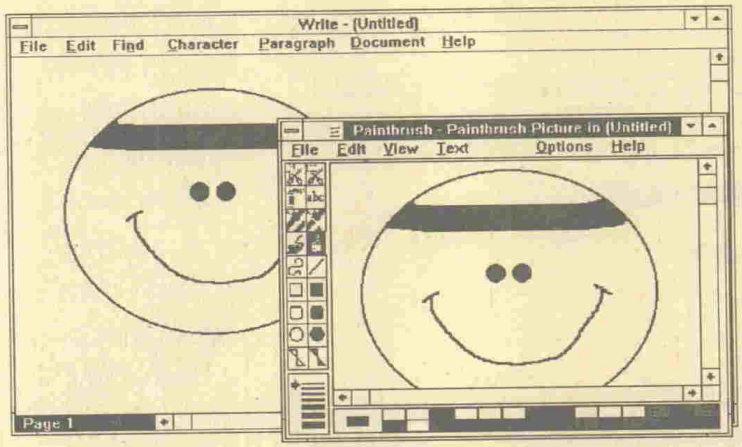
ตอนนี้ออปเจ็กต์ที่เราวาดรูปไว้ก็จะปรากฏในเอกสารของโปรแกรม Write ดังรูปที่ 5 ซึ่งเป็นการตรึงออปเจ็กต์บนเอกสารนั้นเอง ถ้ายังไม่เห็นออปเจ็กต์ให้กดคีย์ PgUp จนกว่าจะเห็น (พื้นที่ของรูปที่เพนต์บรัชสร้างขึ้นอาจจะใหญ่เกินส่วนที่เราวาดไปหน่อย) จากนี้ไปรูปที่วาดนี้ก็จะกลายเป็นส่วนหนึ่งของเอกสาร Write แล้ว เราสามารถแก้ไขออปเจ็กต์นี้ได้โดยการดับเบิ้ลคลิ๊กบนออปเจ็กต์ที่นำมาตรึงไว้ โปรแกรม Paintbrush ที่ใช้วาดรูปออปเจ็กต์ก็จะเปิดขึ้นพร้อมกับแสดงออปเจ็กต์ดังรูปที่ 6 หรือจะแก้ไขออปเจ็กต์โดยเรียกจากคำสั่ง Edit Paintbrush Picture Object บนเมนู Edit ก็ได้ เมื่อเราแก้ไขเสร็จแล้วก็ให้เลือกคำสั่ง Update เพื่อปรับปรุงข้อมูลที่แก้ไข แต่ถ้าเราออกจากแอพพลิเคชันเซอร์ฟเวอร์โดยยังไม่ได้มีการปรับปรุง ส่วนใหญ่ตัวเซอร์ฟเวอร์จะถามเราก่อนว่าจะทำการปรับปรุงก่อนกลับไปยังแอพพลิเคชันไคล์เอ็นต์เดิมหรือไม่ ให้เราเลือก Yes ถ้าต้องการปรับปรุง หรือเลือก No ถ้าไม่ต้องการปรับปรุง

เลือกรูปแบบที่จะแปะ

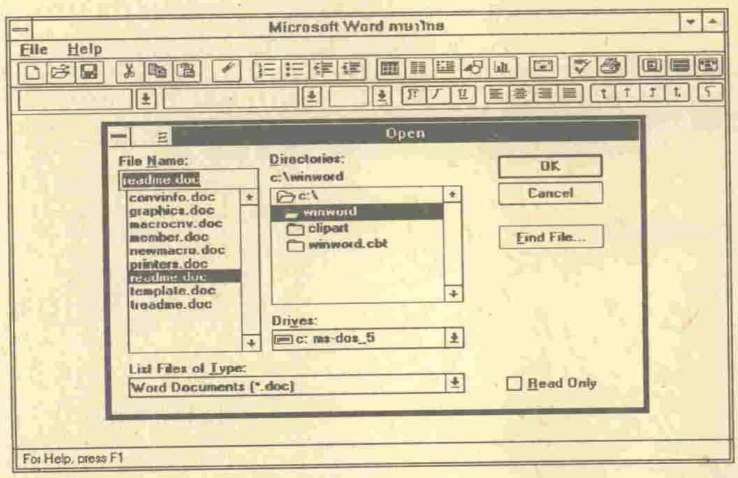
ถ้าเราก๊อบปี้จากแอพพลิเคชันที่เป็นเซอร์ฟเวอร์ไปยังคลิปบอร์ด แล้วจากนั้นแปะออปเจ็กต์ไว้ในแอพพลิเคชันที่เป็นไคล์เอ็นต์ อาจจะไม่ได้ผลตามที่เราคาดไว้เสมอไป ยกตัวอย่างเช่น สมมติว่าเราใช้คำสั่ง Copy บนเมนู Edit ในเวิร์ดฟอร์วินโดวส์ 2.0 เพื่อก๊อบปี้ข้อความไปยังคลิปบอร์ด จากนั้นเรากลับไปยังโปรแกรม Write และใช้



รูปที่ 5 ออปเจ็กต์ของรูปที่วาดจะถูกตรึงไว้บนเอกสารของ Write



รูปที่ 6 เมื่อสั่งแก้ไขออปเจ็กต์ แอพพลิเคชันที่ใช้สร้างออปเจ็กต์จะถูกเปิดขึ้นพร้อมแสดงออปเจ็กต์เพื่อให้แก้ไข



รูปที่ 7 ลองโหลดไฟล์ readme.doc ในเวิร์ดฟอร์วินโดวส์

คำสั่ง Paste บนเมนู Edit แทนที่จะเป็นการแปะข้อความดังที่เราคาดเอาไว้ กลับกลายเป็นแปะรูปไอคอนซึ่งเป็นแพคเกจที่เก็บข้อความแทนก็ได้ ซึ่งผลที่ได้ขึ้นอยู่กับรูปแบบและลำดับข้อมูลที่อยู่ในคลิปบอร์ด

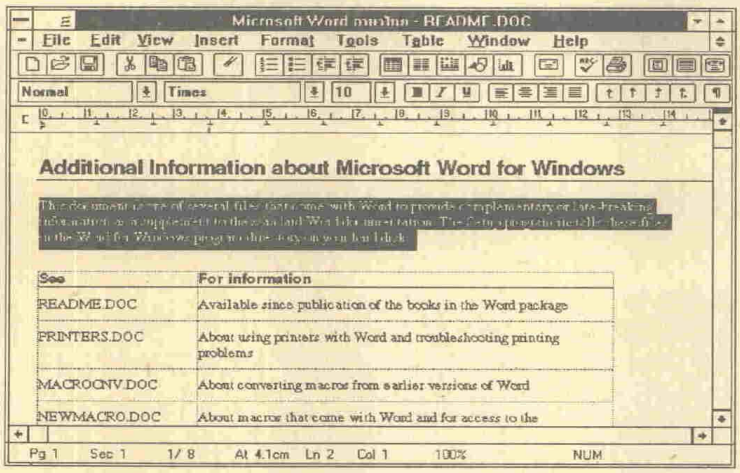
โดยปกติแล้วแอพพลิเคชันที่เป็นเซอร์ฟเวอร์มักจะใส่ข้อมูลในคลิปบอร์ดไว้หลายรูปแบบ การควบคุมรูปแบบข้อมูลที่จะใช้ในการแปะข้อมูลนั้นเราสามารถใช้คำสั่ง Paste Special บนเมนู Edit ในแอพพลิเคชันที่เป็นไคล์เอ็นต์แล้วเลือกรูปแบบข้อมูลที่จะแปะลงไป ลองดูตัวอย่างต่อไปนี้เป็นการก๊อบปี้ข้อความจากเวิร์ดฟอร์วินโดวส์ไปยังโปรแกรม Write โดยเลือกรูปแบบข้อมูลที่ต่างไปจากปกติ

1. ลองเปิดไฟล์ readme.doc ในเวิร์ดฟอร์วินโดวส์ดูรูปที่ 7 โดยใช้คำสั่ง Open บนเมนู File
2. เลือกบริเวณข้อความที่จะก๊อบปี้โดยการแดร็กบริเวณข้อความดังรูปที่ 8
3. เลือกคำสั่ง Copy บนเมนู Edit
4. ย้ายการทำงานไปที่โปรแกรม Write
5. บนเมนู Edit เลือก Paste Special จะปรากฏไดอะล็อกบ็อกซ์ดังรูปที่ 9 จะเห็นรูปแบบข้อมูลที่มีให้เลือกในคลิปบอร์ด
6. เลือกรูปแบบ Word Document Object
7. เลือกปุ่ม Paste

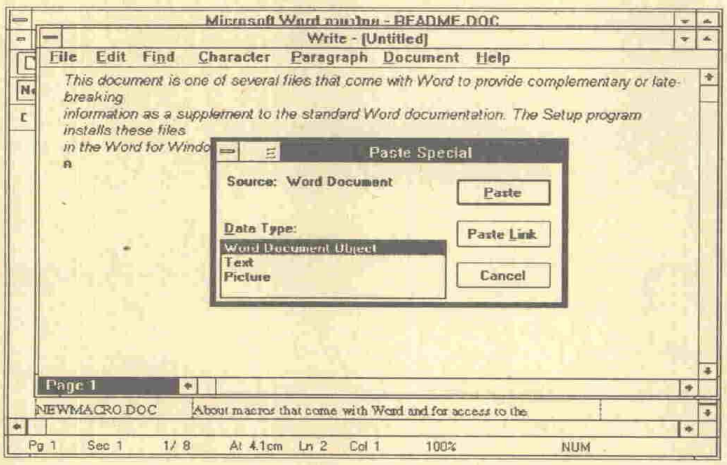
ข้อมูลรูปแบบออปเจ็กต์ (รูปไอคอนของเวิร์ดฟอร์วินโดวส์) จะถูกแปะลงในเอกสารของโปรแกรม Write แทนที่จะเห็นเป็นข้อความ จะเห็นว่าการเลือกรูปแบบข้อมูลที่ต่างกันก็จะให้ผลที่ต่างกันไป รูปแบบในลักษณะออปเจ็กต์นี้ ตัวมันจะเก็บข้อความไว้ เราสามารถดับเบิ้ลคลิ๊กเพื่อแก้ไขข้อความได้เช่นกัน ซึ่งแอพพลิเคชันเซอร์ฟเวอร์จะถูกโหลดเพื่อให้เราแก้ไขข้อความในออปเจ็กต์ที่ตรึงไว้นี้

สรุปคำสั่ง

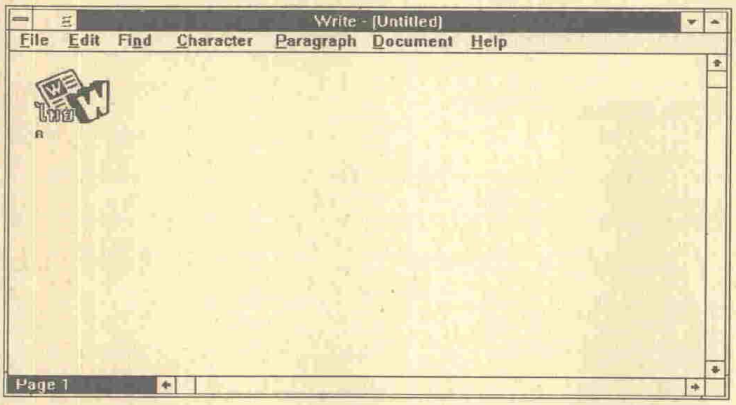
สรุปคำสั่งที่เกี่ยวข้องกับ OLE ซึ่งเรามักพบในแอพพลิเคชันที่เป็นไคล์-



รูปที่ 8 เลือกบริเวณข้อความที่จะก๊อบปี้ของเอกสารเวิร์ดฟอร์วินโดวส์เพื่อนำไปแปะไว้ในโปรแกรม Write



รูปที่ 9 เลือกรูปแบบข้อมูล Word Document Object ที่จะแปะในเอกสาร Write



รูปที่ 10 รูปแบบข้อมูลออปเจ็กต์ที่แปะ จะปรากฏเป็นไอคอนแทนที่จะเป็นข้อความ

เอ็นต์ดังนี้

– Edit Insert Object เป็นการเปิดเซอร์ฟเวอร์และอนุญาตให้เราเลือกหรือสร้างออปเจ็กต์ในการตรึง

– Edit Object เปิดแอพพลิเคชันเซอร์ฟเวอร์ที่ใช้สร้างออปเจ็กต์ที่ตรึงไว้เพื่อที่ว่าเราจะได้แก้ไขมัน อีกทางหนึ่งคือ เราสามารถดับเบิ้ลคลิ๊กบนออปเจ็กต์ในแอพพลิเคชันไคล์เอ็นต์เพื่อให้เกิดผลเช่นเดียวกันนี้ได้เช่นกัน

– Edit Paste แปะออปเจ็กต์เมื่อข้อมูลบนคลิปบอร์ดได้มาจากเซอร์ฟเวอร์

– Edit Paste Special อนุญาตให้เราเลือกรูปแบบข้อมูลบนคลิปบอร์ดที่จะแปะซึ่งได้มาจากเซอร์ฟเวอร์

คำสั่ง OLE ที่มักพบในแอพพลิเคชันที่เป็นเซอร์ฟเวอร์มีดังนี้

– File Update ปรับปรุงข้อมูลที่เปลี่ยนแปลงในแอพพลิเคชันไคล์เอ็นต์ที่เราตรึงไว้ ปกติคำสั่งนี้จะแทนที่คำสั่ง Save บนเมนู File

– File Exit & Return to ปิดแอพพลิเคชันเซอร์ฟเวอร์และกลับไปยังแอพพลิเคชันไคล์เอ็นต์ ที่เรียกเซอร์ฟเวอร์นี้ คำสั่งนี้จะแทนที่คำสั่ง File Exit ชื่อแอพพลิเคชันที่จะกลับไปสังเกตหลังคำ Exit & Return

– Edit Copy/Cut ก๊อบปี้หรือตัดข้อมูลจากเซอร์ฟเวอร์ไปยังคลิปบอร์ดซึ่งมีหลายรูปแบบข้อมูล

ก่อนจาก

สำหรับตอนคงจะพอเข้าใจเรื่องเกี่ยวกับ OLE และคำสั่งที่เกี่ยวข้องกับแอพพลิเคชันกันบ้างแล้ว ซึ่งเรื่องของ OLE หลักของมันก็คือการตรึงข้อมูลในแอพพลิเคชันที่ต่าง ๆ กัน เพื่อให้สามารถแสดงหรือแก้ไขข้อมูลโดยเรียกจากแอพพลิเคชันเดียวกันนั้นได้เลย โดยไม่ต้องออกจากแอพพลิเคชัน ทำให้เอกสารของเรามีความหลากหลายของข้อมูลมากขึ้น